

EFFICIENT LOCKING FOR THREAD-SAFE SELF-MODIFYING CODE

ABSTRACT OF THE DISCLOSURE

A locking mechanism for use in a multi-thread environment supporting self-modifying code in which modifications to the code are made at runtime. The locking mechanism having associated helper code accessed by a call from the first instruction address in the code block. The helper code calculating the binary encoding for the call instruction and using an atomic compare and exchange instruction to compare the calculated binary encoding with the actual contents of the first instruction address. Where there is a match, a self loop instruction is written to the first instruction address to lock the specified code block for subsequent threads. The helper code contains instructions to resolve the references in the specified block. The last such instruction is an atomic store operation to replace the self loop instruction at the first instruction address with the appropriate modified instruction.